LA-UR-03-5939

Approved for public release; distribution is unlimited.

Title: Polygonal Surface Mesh Quality Improvement

Author(s): Rao V. Garimella, Mikhail J. Shashkov
T-7, LANL

Submitted to: Presentation at 4th Symposium on Trends in Unstructured Mesh Generation, Albuquerque, NM July 28-30, 2003

Polygonal Surface Mesh Improvement (LA-UR-03-5939)

Rao Garimella, Mikhail Shashkov

T-7, Mathematical Modeling and Analysis,
Los Alamos National Laboratory,
Los Alamos, NM.

Introduction

- Surface mesh - set of non-overlapping, tiling polygons approximating a smooth surface in $\Re^3$

- Surface mesh quality important for meshing and analysis

- Surface mesh quality - Mesh Size Gradation, Element Shape (e.g., condition number shape measure)

- Effects of poor surface mesh quality
  - ◇ May lead to failure of volume meshing algorithm
  - ◇ Causes poorer quality of volume elements
  - ◇ Influences accuracy of numerical simulations

Improvement of Surface Mesh Quality

- Can improve element quality and mesh gradation by:
  - ◇ Local mesh modifications like edge split, edge swap, etc. (triangular meshes only)
  - ◇ repositioning nodes or smoothing (useful for all meshes).
- Focus on element shape improvement by node repositioning
- Must preserve surface and mesh characteristics during improvement
- Minimizing surface and mesh changes important for:
  - ◇ Preservation of forces like surface tension
  - ◇ Accuracy of solution transfer between meshes

Surface Quality Optimization w.r.t. Parametric Coordinates

- Minimize change to surface characteristics by constraining nodes to
  - ◇ Smooth surface underlying surface mesh, or
  - ◇ Discrete surface formed by faces of surface mesh
- Common to constrain nodes to surface by repositioning in 2D parametric space of surface
- Global parametric space usually unavailable for discrete surfaces
- Global parametric space construction can be expensive

Surface Quality Optimization Using Local Parametrization

- Reposition nodes in LOCAL instead of global parametric space
- Local parametric space - barycentric mapping of triangular element or triangular facet of element
- Keep track of original mesh element and triangular facet in which node is moving
- Vertex in surface interior - containing element is mesh face
- Vertex on surface boundary - containing element is boundary mesh edge
- If node moves out of element, switch to parametric space of adjacent element

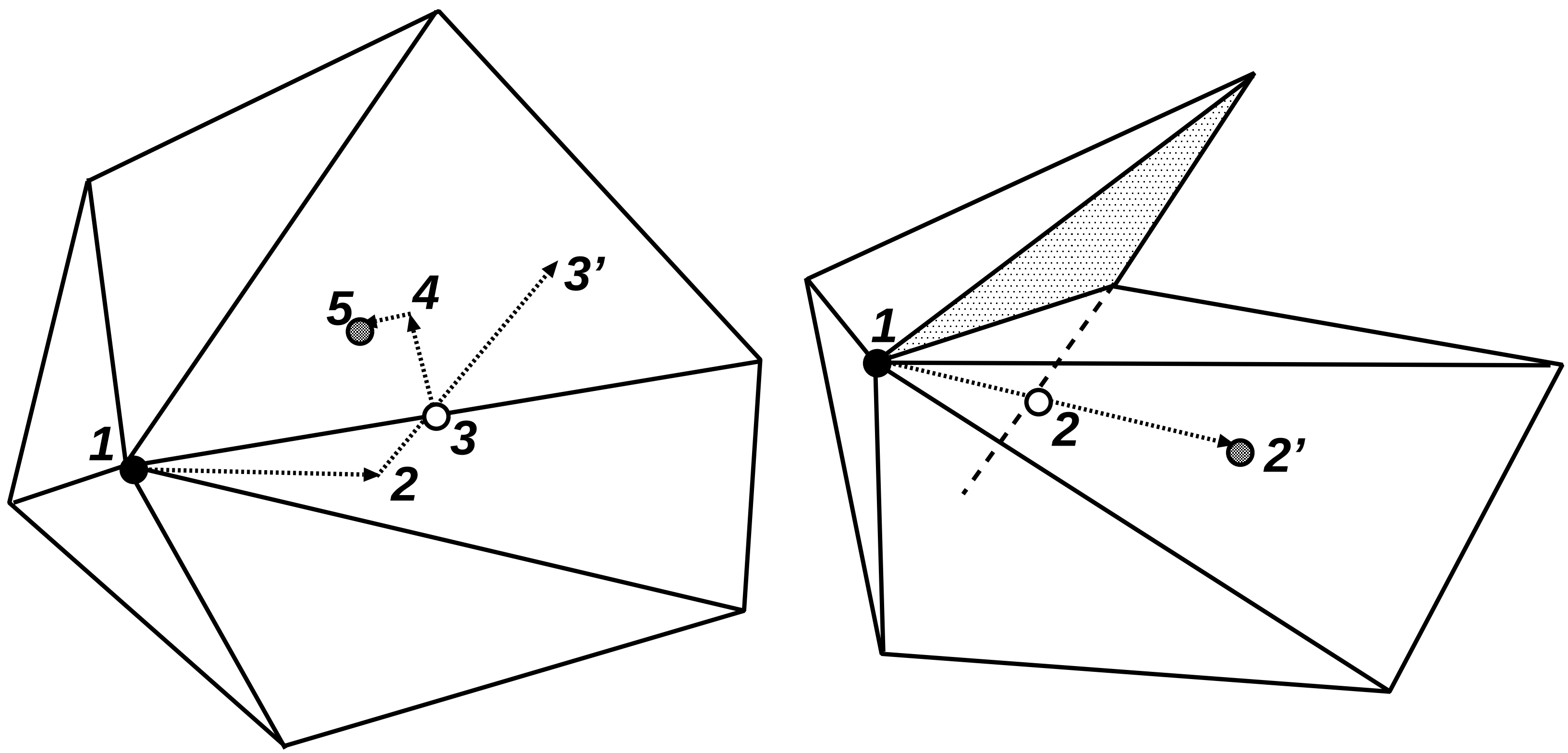
Optimization w.r.t. Parametric Coordinates

- **Consider Gradient Based Optimization of Surface Mesh Quality Function ϕ**
- $\phi = \phi(\mathbf{x}_i)$ **and** $\mathbf{x}_i = f(\mathbf{s}_i)$**, where**
  - ◇ $\mathbf{x}_i$ **are real coordinates of node** i**,**
  - ◇ $\mathbf{s}_i$ **are parametric coordinates of node in containing element.**
- **Function value at given parametric location,** $\mathbf{s}_i$**, is computed by** $\phi(f(\mathbf{s}_i))$
- **Gradient w.r.t. parametric coordinates,** $\mathbf{s}_i$**, evaluated numerically**
- **Line search for minimum is conducted along gradient direction in local parametric space**

Line Search or 1D Minimization

- **Find distance, α, along gradient direction, d, so that**
  - **Objective function is minimized, or**
  - **Line search constraints are encountered**
- **Line Search Constraints:**
  - **Parametric Bounds: If parametric bounds are violated, vertex moves out of the containing element and off the surface**
  - **Mesh Validity: Large movement along search direction makes some connected elements invalid**
- **Algorithm uses incremental stepping with step size control**

Line Search Constraints



(a) Parametric Bounds Constraint **(b) Mesh Validity Constraint**

Parameter Update and Parameterization Change

- Using step size α from line search, update parameters
- $\mathbf{s}_{new} = \mathbf{s}_{old} + \alpha\mathbf{d}$
- If line search stops at minimum or stops due to mesh invalidity

  - continue optimization with new gradient calculation
- If line search stops at parametric bounds

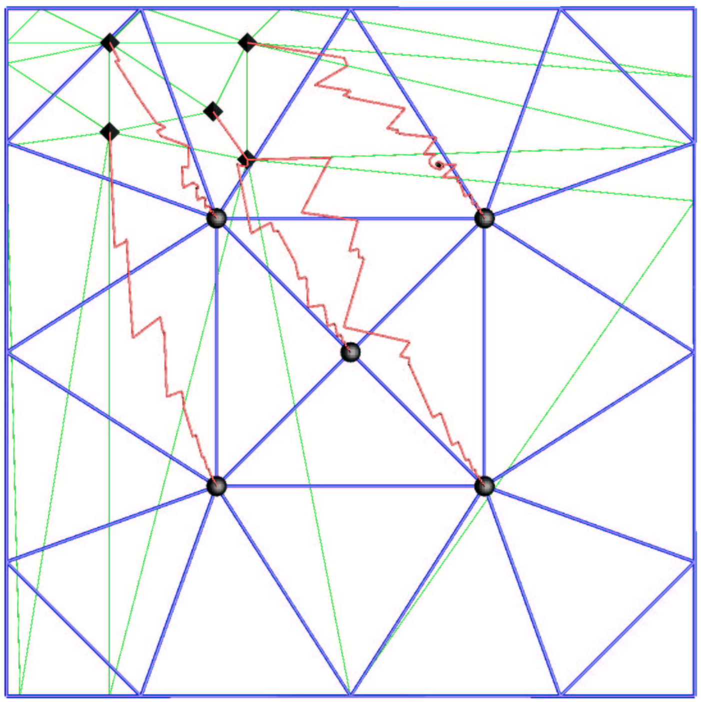
  - restart optimization in parametric space of adjacent element/facet
- If search switches too much between two faces/facets

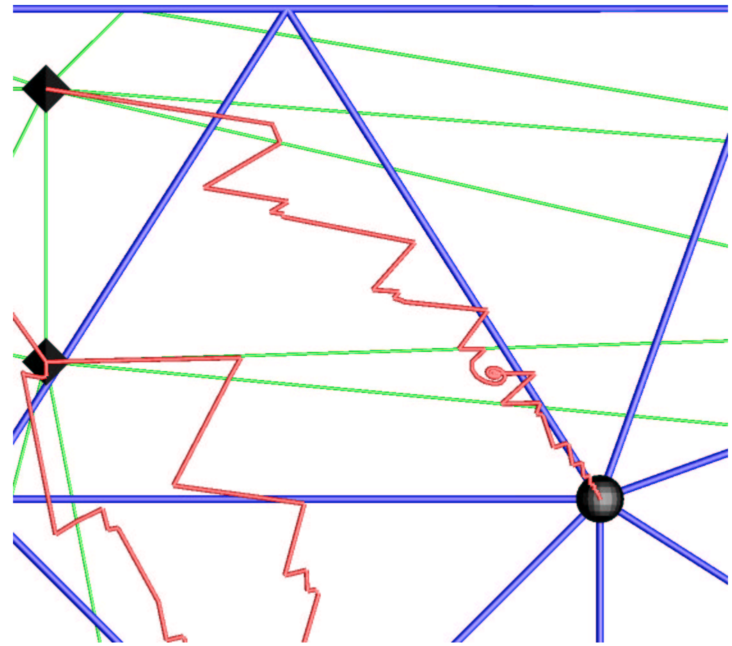
  - proceed along common edge

Optimization of Global Function by Local Iterations

- Desirable to reposition all mesh vertices simultaneously by minimizing global function
- However, local parametric bounds impose too strong a constraint
- Line search seeks single step size for movement of all vertices
- Even if one vertex goes out of parametric bounds, line search must halt
- So, reposition vertices one at a time by minimizing a local piece of global objective function
- Iterate over all vertices until vertex movement is minimal

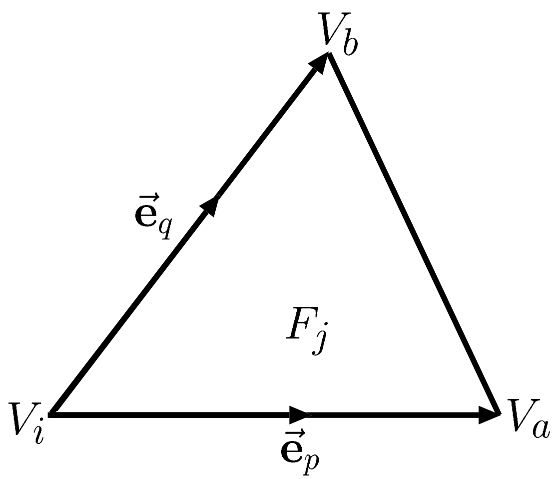
Illustration of Vertex Movement





Minimization of Global Condition Number Objective Function (described later)

Condition Number (CN) Measure of Element Quality



$\mathbf{J}_{ji} = [\vec{\mathbf{e}}_p \mid \vec{\mathbf{e}}_q]$ = **Jacobian of** F_j **at** V_i

$$\boxed{\kappa(\mathbf{J}_{ji}) = \frac{l_p^2 + l_q^2}{A_j}}$$ = **CN of** $\mathbf{J}_{ji}$ **in** $\Re^2$

$$l_p = |\vec{\mathbf{e}}_p|\,, \quad l_q = |\vec{\mathbf{e}}_q|$$

$$A_j = (|\ \mathbf{e}_p \times \mathbf{e}_q\ |)\,/2$$

= **Area of** $\triangle$ **formed by** $\mathbf{e}_p, \mathbf{e}_q$

κ **- function of triangle lengths; rotation invariant**

Therefore, κ **for** $\Re^2$ **useful for measuring triangle quality in** $\Re^3$

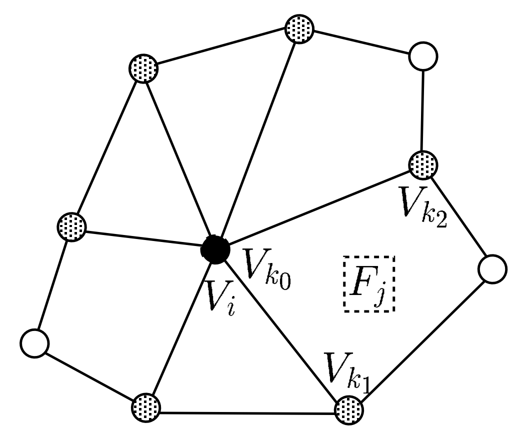
Local Condition Number (CN) Objective Function

- **Define ψ_i^c as sum of all CNs involving vertex V_i**
- **ψ_i^c is sum of CNs at V_i and its adjacent vertices in each face, F_j connected to V_i**
- **Minimization of ψ_i^c smooths distribution of angles and edge lengths around V_i**

$$\psi_i^c = \sum_j \sum_k \kappa(\mathbf{J}_{jk}) = \sum_j \sum_k \frac{(l_p^2)_{jk} + (l_q^2)_{jk}}{A_j}$$

$$j \in \{j \mid F_j \in \{F(V_i)\}\}$$

$$k \in \{k \mid V_k \in \{\{V(F_j)\} \cap \{V(\{E(V_i)\})\}\}$$



Condition Number based Optimization

- **GLOBAL CN function** $\boxed{\Psi^c = \sum_i \psi_i^c}$
- **Global function is sum of local functions over all vertices**
- **Minimize global function, Ψ^c, by minimizing local function, ψ_i^c, at each vertex, V_i**
- **Local minimization by non-linear conjugate gradient method**
- **Optimization performed w.r.t. local parametric coordinates**
- **Multiple iterations over all mesh vertices**
- **Process converged if no vertex moves significantly**

Preservation of Mesh Characteristics

- **CN optimization allows vertices to move as much as necessary on the original mesh faces to minimize Ψ^c**
- **Sometimes, local refinement or anisotropy of mesh must be preserved for solution accuracy**
- **Also, improved mesh must be close to original in some applications**
- **Necessary for accuracy of solution transfer between meshes**
- **Examples:**

  - ALE simulations of multi-material gas dynamics

  - Simulation of metal forming processes

Reference Jacobian based Optimization

- **Use Reference Jacobian based optimization for improving mesh and keeping it close to original mesh**
- **STEP I: Optimize Local Condition Number based function**
- **STEP II: Optimize Global Reference Jacobian based function**

Reference Jacobian based Optimization - Step I

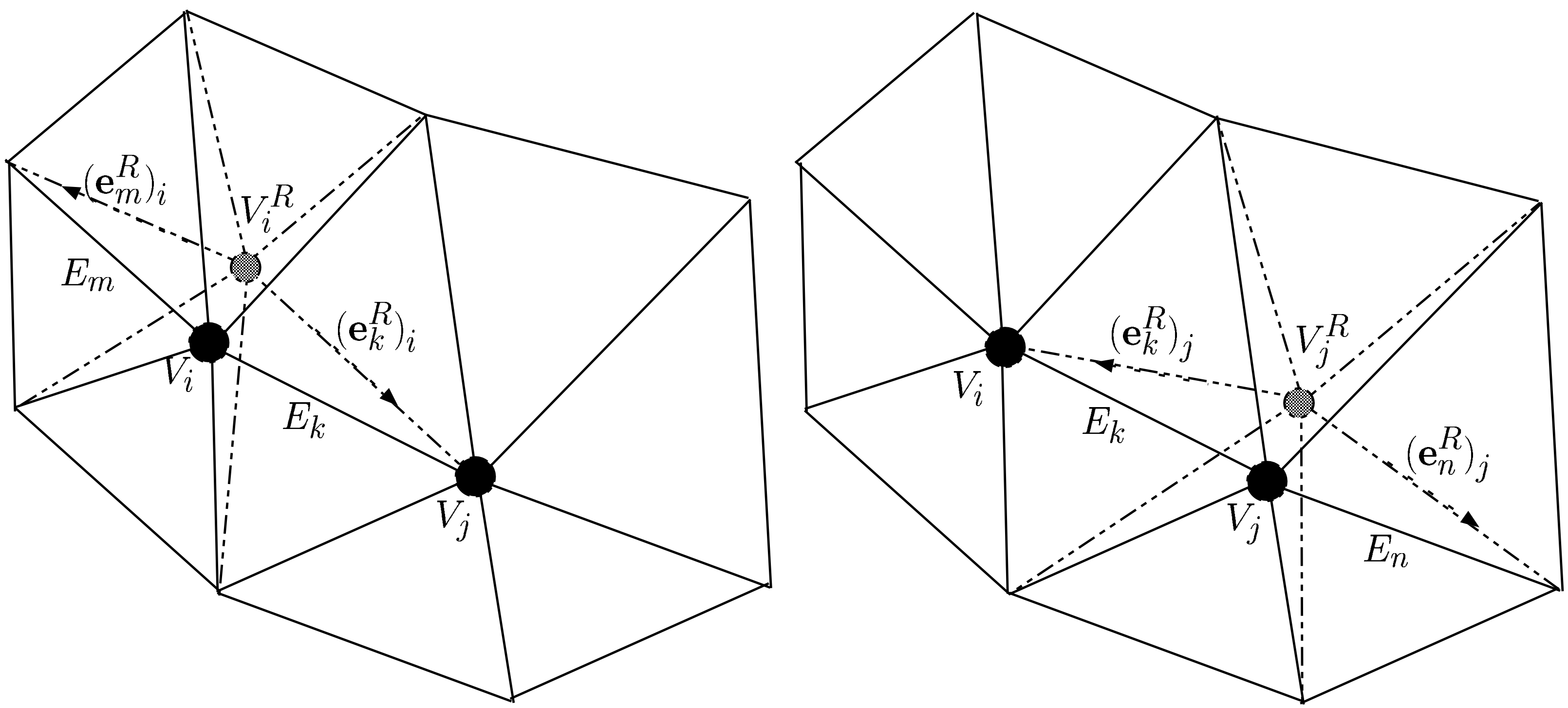
- **Local optimization performed with CN function** ψ_i^c
- **Locally optimal position stored for use in Step II**
- **Vertex not moved to locally optimal position**

Reference Positions, Edges and Jacobians

- **Locally optimal position is stored as REFERENCE POSITION**
- **Two REFERENCE EDGES formed for each real edge**
- **Each reference edge uses real position at one end and reference position at the other**
- **REFERENCE JACOBIAN MATRIX, $\mathbf{J}^R$:**

  - Jacobian matrix formed by pair of reference edges

Reference Positions and Reference Edges



Reference Jacobian (RJ) based Optimization - Step II

- **Goal: Find mesh configuration that achieves compromise between reference edge pairs**
- **Each reference edge has one vertex at optimal location and one at the original location**
- **Therefore, final compromise mesh configuration:**
  - ◇ **improves element quality**
  - ◇ **is close to original mesh**

Global Reference Jacobian (RJ) Objective Function

Global RJ Objective Function Ψ^R:

$$\Psi^R = \sum_i \sum_j \frac{\|\mathbf{J}_{ji} - \mathbf{J}^R_{ji}\|^2}{A_j / A^R_{ji}},$$

$$i \in \{i \mid V_i \in \{V\}\},\ j \in \{j \mid F_j \in \{F(V_i)\}\}$$

$\|.\|$ **is the Frobenius Norm,**

J^R_{ji} **is Reference Jacobian Matrix of** F_j **of** V_i**,**

A^R_{ji} **is area of** $\triangle$ **formed by reference edge vectors at** V_i^R **in** F_j

Local part of Global RJ Objective Function

Define ψ_i^R as part of Ψ^R involving real or reference position of V_i

$$\psi_i^R = \sum_j \sum_k \frac{\|J_{jk} - J_{jk}^R\|^2}{A_j / A_{jk}^R},$$

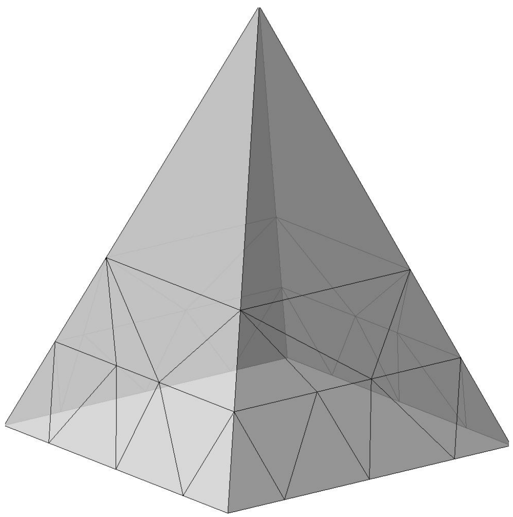
$$j \in \{j \mid F_j \in \{F(V_i)\}\},\ k \in \{k \mid V_k \in \{\{V(F_j)\} \cap \{V(\{E(V)\})\}\}$$

Outer sum is over all faces, F_j, connected to V_i

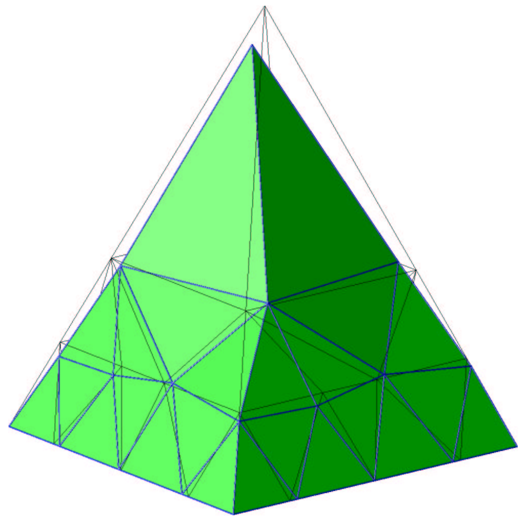
Inner sum is over V_i and vertices of face, F_j, connected to V_i

Iterate over all vertices, performing local optimizations until no further vertex movement is possible

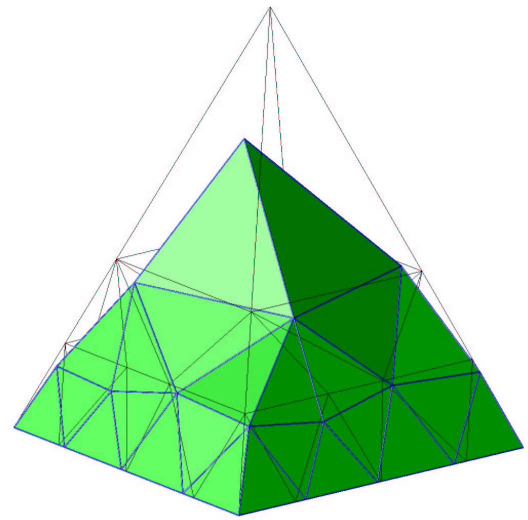
Illustrative Example



Original mesh



RJ Optimization

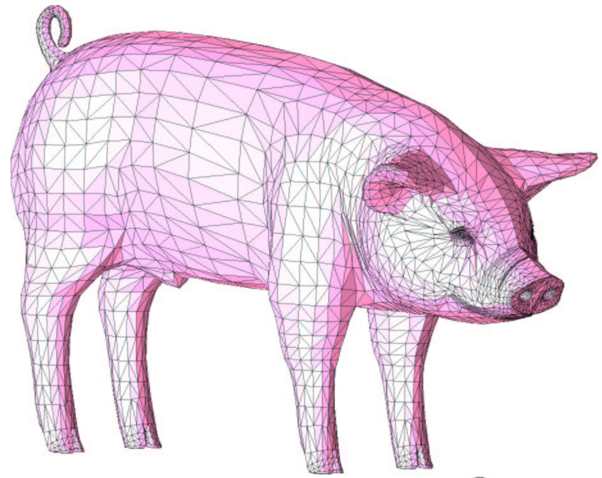


CN Optimization

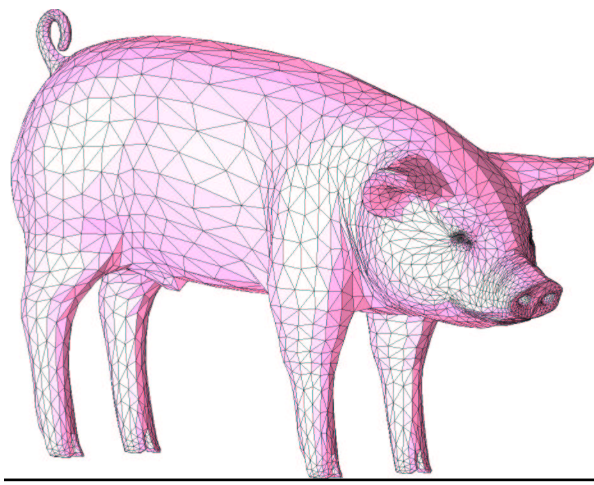
All points still on original mesh

CN optimization moves points more than RJ optimization

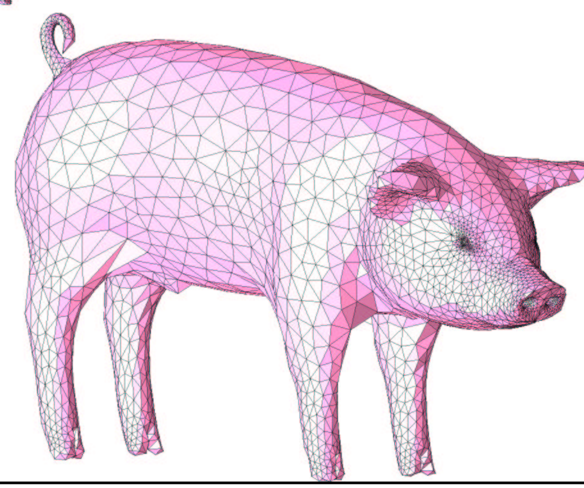
Example: Pig (Computer Graphics Group, U of VA



Original Mesh



RJ Opt. Mesh



CN Opt. Mesh

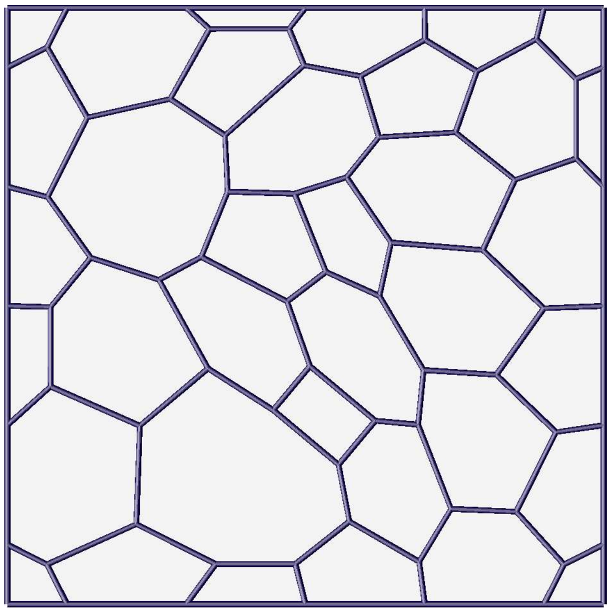
Mesh and Surface Statistics for Pig

$\mathcal{K}_{av}$ = **Normalized mean of condition numbers at vertices of a face**

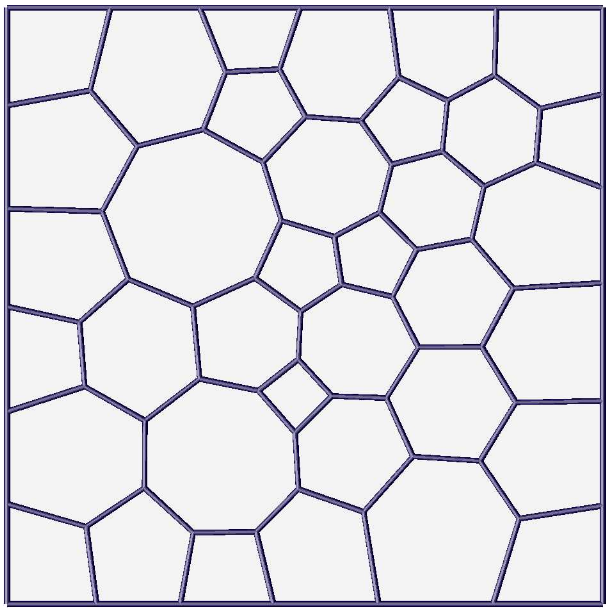
$\mathcal{K}_{av}$			Original	RJO	CNO
1.0	–	1.5	3921	5124	6830
1.5	–	2.0	1734	1257	156
2.0	–	3.0	917	525	48
3.0	–	4.0	247	100	3
4.0	–	5.0	102	22	0
5.0	–	10.0	104	8	3
10.0	–		15	4	0

Surface Distortion Measure	RJ Opt.	CN Opt.
Hausdorff Distance (% of prob. size)	0.6	2.7
Max. Node Movement (% of prob. size)	3.1	11.14
Ave. Node Movement (% of prob. size)	0.3	1.7

Effect of CN Optimization on 2D Polygonal Mesh

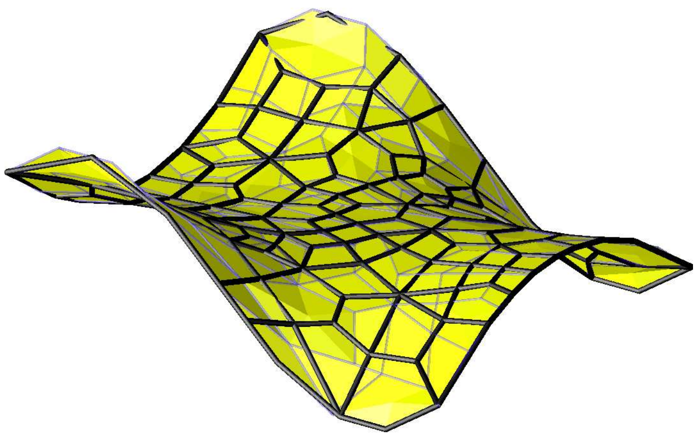


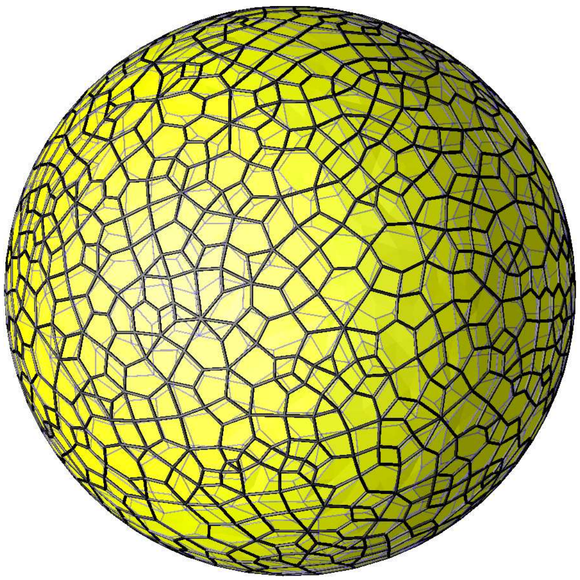
Original Mesh



CN Opt. Mesh

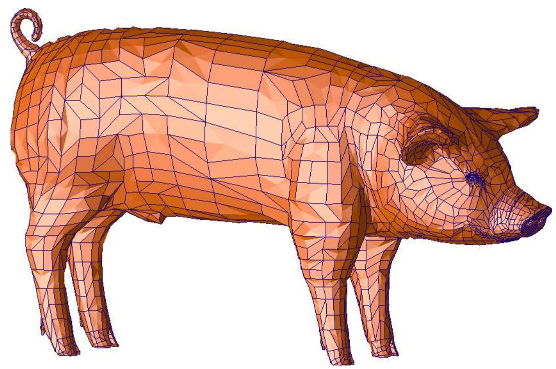
Fidelity of Surface Representation



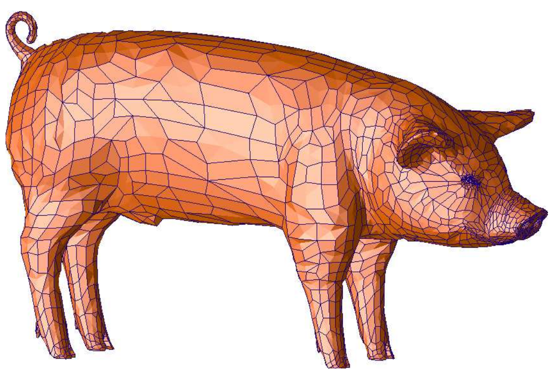


CN optimized mesh (black edges) overlayed on orginal mesh
Hausdorff distance between meshes (to be computed)

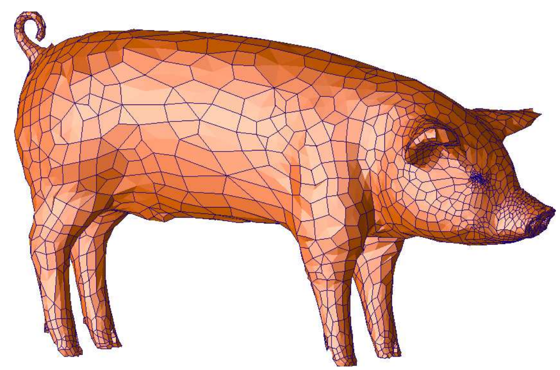
Example: Polygonal mesh of pig



Original Mesh



RJ Opt. Mesh



CN Opt. Mesh

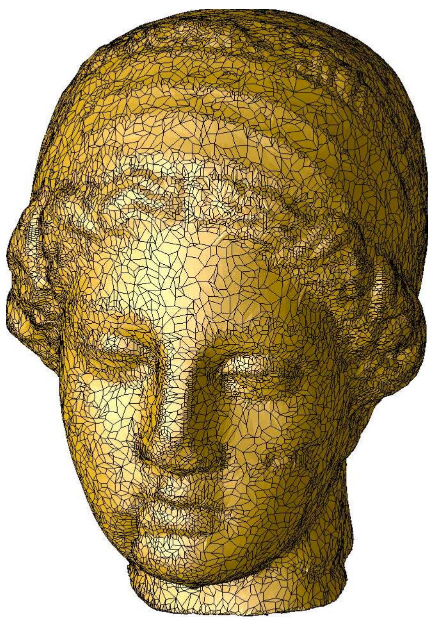
Mesh Statistics for Polygonal Mesh of Pig

$\mathcal{K}_{av}$ = Normalized mean of condition numbers at vertices of a face

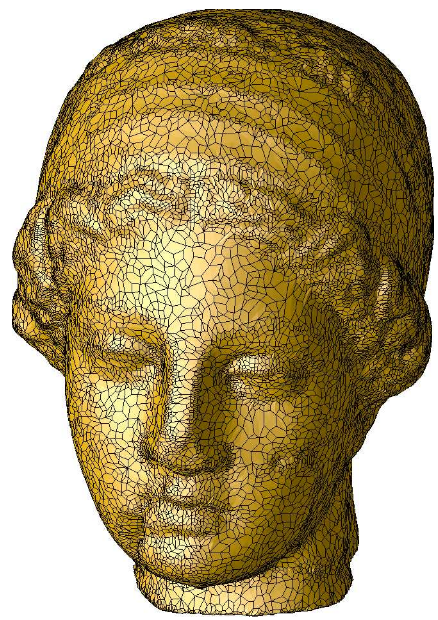
$\mathcal{K}_{av}$			Original	RJO	CNO
1.0	–	1.5	1100	1774	2666
1.5	–	2.0	1017	851	306
2.0	–	3.0	736	360	48
3.0	–	4.0	113	33	6
4.0	–	5.0	25	6	1
5.0	–	7.5	21	4	0
7.5	–	10.0	11	0	1
10.0	–	15.0	3	1	1
15.0	–		3	0	0

Maximum Condition Number before Optimization: 45.07
Maximum Condition Number after RJ Optimization: 14.35
Maximum Condition Number after CN Optimization: 11.44

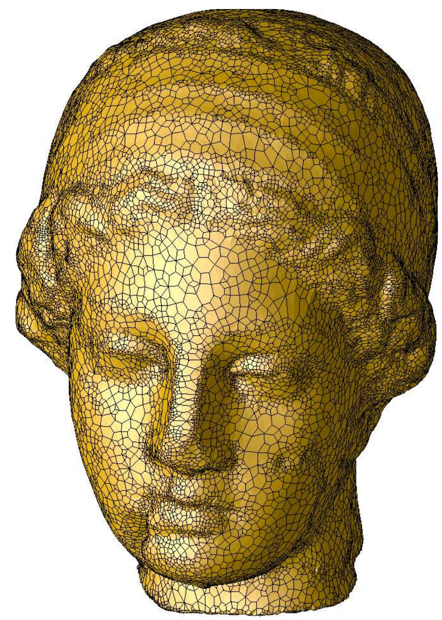
IGEA face model, (Cyberware, Inc.)



Original Mesh



RJ Optimization



CN Optimization

Mesh and Surface Statistics for Igea artifact

$\mathcal{K}_{av}$			Original	RJO	CNO
1.0	–	1.5	29572	37432	39764
1.5	–	2.0	7325	2371	277
2.0	–	3.0	2683	232	0
3.0	–	4.0	335	5	1
4.0	–	5.0	64	1	0
5.0	–	10.0	59	1	0
10.0	–		4	0	0

Surface Distortion Measure	RJ Opt.	CN Opt.
Hausdorff Distance (% of prob. size)	0.2	0.5
Max. Node Movement (% of prob. size)	1.3	3.0
Ave. Node Movement (% of prob. size)	0.2	0.4

Conclusions

- Optimization procedure to improve quality of surface meshes by node repositioning
- CN optimization improves mesh quality (Jacobian condition number) as much as possible
- RJ optimization improves mesh quality but also keeps nodes close to original locations
- Nodes repositioned in series of local parametric spaces to minimize change to surface characteristics
- Barycentric parametrization for triangles
- Triangular facetization of quads and higher polygons
- Procedures tested successfully for complex polygonal meshes
- Future work will make improvements to better handle polygons